WEST Search History

DATE: Sunday, November 30, 2003

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,PGPB,JPAB,EPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
L31	L30 and l25	15	L31
L30	L29 and @AD<19990611	76	L30
L29	device near8 (broadcast or broadcasting) near8 (network adj6 packet)	119	L29
DB=USPT,PGPB; PLUR=YES; OP=ADJ			
L28	L27 and l19	6	L28
L27	L26 and @AD<19990611	61	L27
L26	L25 and l17	121	L26
L25	device near8 (activate or activated or activating or pluging or pluged or plugging or plugged)	72811	L25
L24	device near8 (when or if) near8 (activate or activated or activating or pluging or pluged or plugging or plugged)	0	L24
L23	device near8 (when or if) near8 (activate or activated or activating)	0	L23
L22	device near5 (when or if) near5 (activate or activated or activating)	0	L22
L21	L20 and @AD<19990611	6	L21
L20	L19 and LDAP	36	L20
L19	(register or registration) same (address near5 device)	6846	L19
L18	L17 and l9	8	L18
L17	trap near4 (monitor or monitoring or management or managing)	1225	L17
L16	l1 and @AD<19990611	1262	L16
L15	l3 and @AD<19990611	17	L15
L14	L13 and l3	2	L14
L13	(verify or verification) near5 packet	1339	L13
L12	L10 and l5	0	L12
L11	L10 and l3	0	L11
L10	(network adj4 (managing or management) adj4 packet) near8 (determine or determining or recognize or recognizing)	8	L10
L9	network adj4 (managing or management) adj4 packet	307	L9
L8	L7 and l5	0	L8
L7	(managing adj3 packet) near8 (determine or determining)	6	L7
L6	(acquire or acquiring)near4 address near4 device near8 (managing adj3 packet)	0	L6
L5	(acquire or acquiring)near4 address near4 device	173	L5

L4	(acquire or acquiring) adj3 address	1375	L4
L3	(universal adj plug adj play) or UPnP	226	L3
L2	(universal adj3 plug adj3 play) or UPnP	253	L2
L1	plug adj3 play	3121	L1

END OF SEARCH HISTORY

WEST

Generate Collection

L2: Entry 8 of 75 File: USPT May 13, 2003

DOCUMENT-IDENTIFIER: US 6564216 B2
TITLE: Server manager

Application Filing Date (1):
19981029

Brief Summary Text (6):
In the past, organizations relied on paper based methods of managing IP addresses in a network. DHCP simplified the management and assignment of IP addresses to clients by eliminating the need for the network administrator to manually configure the network. With DCHP, when a client requests an IP address in order to communicate over the network, a DHCP server answers the request by providing network configuration information that was obtained from a database and dynamically assigning an IP address to the client. Each DHCP server manages a range of client addresses and can pick any address from that range as long as it is not being used by a another client managed by that DHCP server. Since the address is dynamically assigned, the client can have a different IP address each time it logs on to the network. Along with the ability to dynamically assign IP addresses, a DHCP server also supports static IP address that have been assigned to one particular client on the network. Based on the configuration information received from the database, the DHCP server can automatically assign a specific IP address to a specific client..sup.1

Brief Summary Text (7):
DNS also simplified the management of networks by translating domain names into IP addresses. Since the DNS server contains a list of domain names and their associated IP addresses, a host or client can be located through by its domain name rather than its IP address. Any given domain name could be associated with more than one IP address and any IP address could be associated with more than one domain name. A DNS server updates the domain name and IP address associations by periodically polling a central database containing configuration information for the network. When a new client is assigned an IP address by a DHCP server, the new configuration information is stored in the central database. Each DNS servers on the network poll the central database for configuration changes. If a new IP address was assigned to a client managed by a DNS server, the DNS server updates the domain name with the new IP address or updates the IP address with the new domain name..sup.2

Detailed Description Text (4):
In the described embodiment, the network includes a server manager 201, one or more DNS servers 202A-N, one or more DHCP servers 203A-N, a central database 204, a NMS 205, one or more clients 208, a Lightweight Directory Access Protocol (LDAP) server 209 and a LDAP database 210. The one or more DNS servers 202A-N and DHCP servers 203A-N are coupled in communication with the server manager 201 through individual communication channels. The server manager 201 is coupled in communication with the central database 204 over a single communication channel 206 and the NMS 205 over a single communication channel 207. The LDAP server 209 is coupled in communication with the server manager 201 over a single communication channel 211. However, in alternative networks, the LDAP server 209 can communicate directly with the central database 204.

Detailed Description Text (11):
Once the DNS servers 202A-N and DHCP servers 203A-N establish a link with the server manager 201, the servers can issue requests for configuration information from the central database 204 or send updated configuration information to the central database 204. The server manager 201 synchronizes all of the requests and updates from the servers and transmits them to the central database 204. The server manager 201 monitors all the DNS servers 202A-N and DHCP servers 203A-N on the network from a single point and acts as a single pipeline to the central database 204. For example, when a new client 208 sends a request for an IP address to a DHCP server 203A, the DHCP server

203A determines if it can send configuration information to the requesting client 208. If the DHCP server 203A can give an IP address and configuration information to the client 208, it sends host configuration information and an IP address to the client 208. The DHCP server 203A automatically registers the new domain name, the IP address and the host configuration information with the central database 204 through the server manager 201. The DNS server 202A detects the new IP address through the server manager and updates its DNS information. When the lease expires or the client 208 leaves the network and releases the IP address, the DHCP server 203A notifies the central database 204 of the change through the server manager 201. The IP address is available for reassignment by the DHCP server 203A to a new client. Therefore, the server manager 201 eliminates the need for the individual DNS servers 202A-N and DHCP servers 203A-N to establish direct communication channels with the central database by providing access to the central database 204 through one communication channel 206. (more????)

Detailed Description Text (20):
The present invention also allows a user to be authenticated and binds the user to the IP address that was given to it by the DHCP server on the network. FIG. 8 provides a flow diagram which illustrates a method of the described embodiment for authenticating a user and binding the user to their current address. First a client 208 requests an IP address from the DHCP server 203N on the network, step 801. The DHCP server 203N dynamically assigns the client 208 an IP address before it has been authenticated, step 802. The client 208 then issues a registration request with the binding server 209 and communicates its userid, password and the IP address it just obtained from the DHCP server 203N to the binding server 209, step 803. The method of communication used by the client in the described embodiment is the Hyper Text Transfer Protocol (HTTP) but alternative methods can be used. Currently the client does not provide itself with a userid or password. However, two examples of methods for the client to obtain its userid and password are to have the user go to a World Wide Web (WWW) page or download a Java applet to obtain information from a PC or workstation which could modify the operating system to automatically provide it. The binding server 209 then authenticates the userid, password and IP address through an LDAP request to the LDAP database 210, step 804. The LDAP request searches the LDAP database 210 for the userid, password and the possible IP addresses that the DHCP server 203N could assign, step 805. The LDAP database 210 is organized in a tree hierarchy. For example, the root of an Internet address is at the top and the common name associated with the user is at the bottom. The LDAP database 210 is accessible through an open, standards based protocol such as TCP. If the information is found in the LDAP database 210, it notifies the binding server 209 that the user credentials were verified by returning the authenticated credentials, step 807..sup.22 The binding server 209 then sends the authenticated credentials to be stored in the central database 204, step 808. In the described embodiment, the binding server 209 communicates with the server manager 201 over a single channel 211 to store the credentials in the central database 204. However, the binding server 209 could also communicate directly or through some other device with the central database 204 in order to store the authenticated credentials.

CLAIMS:

1. A server manager configured to receive a plurality of requests for network protocol address information from one or more network protocol address management servers to prioritized said plurality of requests, and to access the requested network protocol address information from a database based on said priority, and configured to automatically and periodically poll the database for updates to said network protocol address information, and to transmit said updates to at least one of said network protocol address management servers.

2. The server manager as recited by claim 1 further configured to synchronize all communication to and from the one or more network protocol address management servers and act as a single pipeline to the database.

5. A server manager configured to periodically poll for updates of network protocol address information from a database, and to communicate said updates to one or more network protocol address servers requiring said updates, and in between consecutive polling for updates, to transmit network protocol address information to a requesting network protocol address server.

8. A method of communicating information between a database and one or more network protocol address management servers through a server manager comprising: receiving at the server manager a plurality of requests for network protocol address information stored in the database and updates thereof from the one or more network protocol

address management servers; prioritizing said plurality of requests for the network protocol address information; communicating by the server manager the requests for the network protocol address information stored in the database and the updates thereof from the one or more network protocol address management servers to the database based on said priority; automatically and periodically polling said database for updates to said network protocol address information; receiving at the server manager the requested information and the updates from the database; and transmitting by the server manager the requested information to the requesting one or more network protocol address management servers and transmitting the updates to one or more network protocol address management servers is affected by said updates.

9. A widely-distributed IP network comprising a database accessible by one or more DNS and DHCP servers; and a server manager: to receive requests for network protocol address information stored in said database from said one or more DNS and DHCP servers; prioritize said requests for said network protocol address information; access the requested network protocol address information from said database based on said priority; and automatically and periodically poll said database for updates to said database.

11. The widely-distributed IP network as recited by claim 9 wherein the server manager controls the database and the DNS and DHCP servers in dynamic configuration of the IP addresses.

12. A server manager: to receive requests for network protocol address information stored in said database from the one or more DNS and DHCP servers; prioritize said requests for said network protocol address information; access the requested network protocol address information from said database based on said priority; and automatically and periodically poll said database for updates to said database.

L2: Entry 51 of 75 File: USPT Feb 8, 2000

DOCUMENT-IDENTIFIER: US 6023464 A
TITLE: Auto-provisioning of user equipment

Abstract Text (1):
A system and method for provisioning a user terminal for accessing a wideband cable data distribution network, such as an Internet-over-cable system, utilizes an auto-provisioning web server to allow provisioning of a user cable modem without need for a field technician to separately input associated modem identification information via a telephone. A LDAP directory is used to store all provisioning information, and is accessible by a DHCP server to selectively allocate network IP addresses to only provisioned terminals.

Application Filing Date (1):
19971223

Brief Summary Text (13):
The present invention utilizes four subsystem operations which include: (1) an "inventory pre-provisioning system" for inputting cable modem serial numbers and MAC addresses into a billing system and into a network user database such as a lightweight directory access protocol (LDAP) directory; (2) a DHCP server functioning to automatically allocate and assign IP addresses to user terminals; (3) an LDAP directory server to manage a plurality of directory subtrees used to provide provisioning of equipment; and (4) an auto-provisioning web server.

Detailed Description Text (3):
A lightweight directory access protocol (LDAP) directory 22 is connected to the DHCP server and functions as a network user information and access privilege database. An LDAP directory server 24 manages a plurality of directory subtrees used to perform auto-provisioning as described below.

Detailed Description Text (4):
A pre-provisioning inventory input system 26 is connected to the system for storing cable modem serial numbers and MAC addresses into the LDAP 22. An auto-provisioning web server 28 is also provided to allow nonprovisioned and nonregistered users limited network access for the purpose of provisioning or registering a user cable modem or computer.

Detailed Description Text (5):
Description of the auto-provisioning operation of the present invention will be made in connection with the flowchart shown in FIG. 2. As shown at a block 100, individual cable modems or other terminal equipment are added to a system's inventory stock database by delivering the modems to an appropriate warehouse location for storage therein. At block 102, each cable modem serial number and/or MAC address is input into the inventory database, and logged into the appropriate billing system and LDAP directory via the inventory pre-provisioning system 26. At block 104, after logging into the directory, the DHCP will assign each new cable modem to an "unregistered cable modem" service class having limited network access privileges, and is not associated with any particular user.

Detailed Description Text (8):
At block 112, the DSR uses the provisioning system to transmit the new user information into the relevant billing system and into the LDAP directory. As noted above, each new user computer is assigned by the DHCP server to an "unregistered computer" service class having limited access privileges. In particular, access is preferably restricted to the auto-provisioning web server 28. Provisioning system 28 then triggers a work order at block 114 to schedule cable modem installation at the user location.

Detailed Description Text (11):
After signing on, the auto-provisioning web server will query the DHCP server directory for user authentication, and will detect the lack of a provisioned cable modem for the user at block 120. The field technician will input the cable modem serial number to the auto-provisioning web server at block 122. The auto-provisioning web server then queries the DHCP server directory for verification of the cable modem serial number, and at block 124, if the number corresponds to an "unregistered cable modem", provisioning of the cable modem will be completed at block 126. The DHCP server stores in the LDAP the association between the cable modem and the new service user information. This association modifies the DHCP service class and consequently the cable modem access options. The server then resets the cable modem (via a simple network management protocol (SNMP)) so that the new cable modem access options are used.

Detailed Description Text (12):
In further accordance with the present invention, the auto-provisioning web server detects at block 128 that a user computer is not provisioned by examining the source IP address of HTTP traffic from the user computer and using DHCP server queries. The field technician selects the user domain name to be associated with the user computer at block 130, and the auto-provisioning web server completes provisioning of the user computer at block 132. The server creates an DHCP directory entry for the user computer, and sets up the LDAP directory association between the computer and the new service user information. The computer is then identified by the proper domain name, and the associated DHCP server service class is updated.

Detailed Description Text (15):
The auto-provisioning web server queries the LDAP directory for user authentication, and at block 204 will detect that the user cable modem is provisioned, but that the user computer is not provisioned. This is detected by examining the source IP address of HTTP traffic from the user computer and using LDAP queries.

Detailed Description Text (17):
The auto-provisioning web server then completes provisioning of the user computer. At block 214, the server creates and/or modifies an LDAP directory entry for the user computer, and sets up the LDAP direction association between the computer and the service user information. The computer is identified by the proper domain name, and the associated DHCP server service class is updated. As noted above, because an "unregistered computer" service class has a limited authorized access period of time, the new computer DHCP access options are quickly enabled. The web server may also reset the cable modem to process the IP address change of the user computer.

Detailed Description Text (19):
As noted above, in the inventory pre-provisioning process, the inventory programming is arranged to create an LDAP directory entry for each cable modem with service class "unregistered cable modem". The DHCP server options and configuration file are set to provide limited access service for auto-provisioning but not full network or Internet access.

Detailed Description Text (20):
If the user cable modem boots onto the system after being properly "inventory pre-provisioned" in the service class "unregistered cable modem," then the DHCP server and user terminal can begin the process of being allocated an IP address to permit access to the network. The IP address range of user computers in the "unregistered computer" service class is preferably within a private address space such as 10.x.x.x. If the user computer boots on the cable plant without first being auto-provisioned, the DHCP server creates an explicit LDAP directory entry for service class "unregistered computer" when the server assigns the computer an IP address. This allows the computer to boot with limited access for auto-provisioning.

Detailed Description Text (21):
If a user computer is replaced by another user computer using the auto-provisioning web server as described in connection with FIG. 3, then the web server is arranged to copy the assigned IP address from the old LDAP directory entry to the new entry, and delete the old entry. This arrangement minimizes domain name changes for DNS.

CLAIMS:

7. The system of claim 6 wherein said database comprises an LDAP directory.

10. A system for provisioning a user terminal to allow access to a wideband cable data distribution network comprising:

an inventory pre-provisioning system for inputting cable modem serial numbers and MAC addresses into a billing system and into a network user database;

a DHCP server arranged to automatically allocate and assign IP addresses to user terminals;

an lightweight directory access protocol (LDAP) directory server arranged to manage a plurality of directory subtrees used to provide provisioning of equipment; and

an auto-provisioning web server arranged to prompt a user for input of the cable modem identification information when a user needs to be provisioned, and provision the modem by storing in the database user identification information to be associated with the modem identification information, wherein the DHCP server is arranged to assign limited network access rights to pre-provisioned modems allowing access to only the auto-provisioning web server until the modem is provisioned.

11. The system of claim 10 wherein the database comprises a lightweight directory access protocol (LDAP) directory.

L2: Entry 55 of 75 File: USPT Dec 28, 1999

DOCUMENT-IDENTIFIER: US 6009103 A
TITLE: Method and system for automatic allocation of resources in a network

Abstract Text (1):
In a broadband cable data network, a method and system for automatically allocating network resources such as IP addresses to control access to the network utilizes at least one DHCP server, and a common network database formed from a LDAP directory for storing respective user configuration parameters, hardware address registration, and current binding information. A DHCP server can add new hardware address registrations to the LDAP using an "unregistered" service class. The DHCP server sends a DHCP reply tailored for unregistered devices, such as by allocating a privately-allocated IP address with no Internet access, or an IP address for a self-provisioning web server. A DHCP server views IP address allocation as indefinite, while a user will view an IP address allocation as having a short duration. Thus, if the IP network configuration does not change, the user terminal will continue to receive the same allocated IP address due to the DHCP server's perception of an indefinite lease. The consistency of the IP addresses simplifies many operational concerns about dynamic addresses, such as minimizing DNS (domain name service) hostname updates, mapping IP addresses to user terminals during security incidents, etc.

Application Filing Date (1):
19971223

Brief Summary Text (12):
To maximize DHCP server flexibility, the DHCP server of the present invention uses standard network protocols where possible, notwithstanding that a standard DHCP server synchronization protocol has not been adapted by the industry. In particular, an LDAP (Lightweight Directory Access Protocol) directory operates as a DHCP server database, thereby permitting tight integration of subscriber personal information with subscriber device information.

Detailed Description Text (3):
A lightweight directory access protocol (LDAP) directory 22 is connected to the DHCP server and functions as a common network user information and access privilege database. An LDAP directory server 24 manages access to the LDAP directory.

Detailed Description Text (10):
In further accordance with the present invention, all DHCP servers obtain respective configuration parameters, hardware address registration, and current binding information from a common network directory. This arrangement significantly simplifies the restart of a crashed DHCP server, and replacement of an entire DHCP server host.

Detailed Description Text (12):
In further accordance with the present invention, an authorized DHCP server can add new hardware address registrations to the common network directory using an "unregistered" service class. If the "unregistered" service class is not defined for a particular source Ethernet, then the DHCP server will not send a DHCP reply. If the "unregistered" service class is defined, then the DHCP server sends a DHCP reply tailored for unregistered devices, such as by allocating a privately-allocated IP address with no Internet access, or an IP address for a self-provisioning web server. An "unregistered" service class may be used for technician laptop Pcs, for current subscribers who have changed Ethernet boards, and for potential subscribers for Internet over cable service demonstrations.

Detailed Description Text (15):
The network directory is preferably implemented as a lightweight directory access protocol (LDAP) directory which is pooled among all DHCP servers via a LDAP server. The

use of an LDAP directory for the DHCP servers provides DHCP server synchronization and tight integration between user personal information such as name, street address, e-mail address, and other contact information, and user terminal information such as hardware address, IP address, and logical location information.

Detailed Description Text (16):
The LDAP directory is updated before a DHCP server sends a response with a new permanent IP address allocation. Distributed DHCP update replicators are used to send asynchronous updates as is generally well understood in the art. If a DHCP server loses its connection to the update replicators, then the DHCP server re-synchronizes using the LDAP directory. If an update replicator loses a connection to a DHCP server, then the update replicator also re-synchronizes using the LDAP directory.

CLAIMS:

1. In a broadband cable data network where information is distributed from at least one DHCP server to a plurality of user terminals, a method for automatically allocating network resources to control access to the network comprising:

generating a discover message from a user wanting to receive allocation of a network resource;

at a DHCP server, detecting user identification information from the discover message to determine whether an address entry is currently stored for the user;

creating an unregistered class entry and assigning a temporary IP address to the user which allows only limited access rights to the network if an address entry does not exist; and

sending an offer message which communicates the temporary IP address to allow limited access to a network user registration service.

L21: Entry 3 of 6 File: USPT Feb 1, 2000

DOCUMENT-IDENTIFIER: US 6021429 A
**** See image for Certificate of Correction ****
TITLE: Network device which maintains a list of device addresses

Application Filing Date (1):
19961118

Brief Summary Text (7):
One conventional solution to the foregoing problem is to provide a centralized address server to maintain a list of device addresses for a LAN. Such a solution is used in the LDAP protocol. This solution, however, introduces an additional file server onto the LAN, thereby increasing both the cost and the complexity of the LAN.

Detailed Description Text (57):
Step S522 determines whether a new list manager (i.e., a list manager not known to NEB 2) is operating on LAN 1 by comparing a stored list manager device address to the device address provided in response to the broadcasted request for the list manager. If, in step S522, the stored device address of the list manager matches the device address provided in response to the broadcasted request, NEB 2 determines that the list manager for LAN 1 has not changed. Therefore, processing proceeds directly to step S523. However, if it is determined in step S522 that the stored device address of the list manager does not match the device address provided in response to the broadcasted request for the list manager, or alternatively that no list manager device address is stored (as would be the case when NEB 2 is initially powered-up), processing proceeds to step S524. In step S524, NEB 2 registers itself with the new list manager by providing its device address to the new list manager via LAN 1. As described above, the list manager adds the device address of NEB 2 to the list of device addresses maintained therein.

WEST

Generate Collection

L31: Entry 2 of 15 File: USPT Feb 1, 2000

DOCUMENT-IDENTIFIER: US 6021429 A
** See image for Certificate of Correction **
TITLE: Network device which maintains a list of device addresses

Application Filing Date (1):
19961118

Detailed Description Text (34):
Briefly, the process steps shown in FIG. 5 define an invention which controls a network device on LAN 1 to operate as a list manager which maintains a list of device addresses for LAN 1, or to operate as a slave which provides a device address of the network device to a list manager over LAN 1. The invention operates from an activated network device, such as NEB 2, to determine whether a list manager is operating on LAN 1, and to control the activated network device to operate as a slave on LAN 1 when it is determined that a list manager is operating on LAN 1. When it is determined that no list manager is operating on LAN 1, the invention controls the activated network device (i.e., NEB 2) to operate as the list manager for LAN 1.

Detailed Description Text (50):
In the foregoing manner, the present invention reduces the period during which more than one list manager is operating on LAN 1 at the same time. It is noted that although the foregoing process is described with respect to resolving contention between two currently operating list managers, the process can be applied equally well when two network devices on a LAN are activated at roughly the same time. Moreover, the same process can be provided for more than two list managers operating on the same LAN.

CLAIMS:

1. A method of controlling a network device on a local area network (LAN) in which plural different network devices communicate over the LAN by transmitting broadcast packets that are not addressed to any one particular network device and by transmitting addressed packets that are addressed to a network device corresponding to a device address contained therein, the network device being controlled to operate as a list manager which maintains a list of device addresses for the LAN, and being controlled to operate as a slave which provides a device address of the network device to a list manager, the method comprising the steps of:

determining whether a list manager is operating on the LAN by transmitting a broadcast packet from the network device to request a list manager and by waiting for a response for a predetermined period of time after the broadcast packet has been transmitted, said determining step determining that no list manager is operating on the LAN if a response to the broadcast packet is not received by the network device after the predetermined period of time, and determining that a list manager is operating on the LAN if a response to the broadcast packet is received by the network device within the predetermined period of time;

controlling the network device to operate as a slave on the LAN when the determining step determines that a list manager is operating on the LAN, including controlling the network device to transmit an addressed packet to the list manager that includes the device address of the network device; and

controlling the network device to operate as the list manager for the LAN when the determining step determines that no list manager is operating on the LAN, including controlling the network device to maintain a list of device addresses and to respond to a requestor's request for device addresses by transmitting an addressed packet containing device addresses to the requestor.

16. A network device on a local area network (LAN) in which plural different network devices communicate over the LAN by transmitting broadcast packets that are not addressed to any one particular network device and by transmitting addressed packets that are addressed to a network device corresponding to a device address contained therein, the network device being controlled, the network device operating as a list manager for the LAN by maintaining a list of device addresses for the LAN, and operating as a slave on the LAN by providing a device address to another device on the LAN operating as the list manager, the network device comprising:

a memory which stores a device address of the network device and process steps for execution by a processor, and which can store the list of device addresses for the LAN;

a LAN interface which interfaces to the LAN, over which communications including broadcast packets and addressed packets are transmitted to and received from the LAN; and

a processor which executes the process steps stored in the memory (1) to determine whether a list manager is operating on the LAN by transmitting a broadcast packet from the network device to request a list manager and by waiting for a response for a predetermined period of time after the broadcast packet has been transmitted, said determining step determining that no list manager is operating on the LAN if a response to the broadcast packet is not received by the network device after the predetermined period of time, and determining that a list manager is operating on the LAN if a response to the broadcast packet is received by the network device within the predetermined period of time, (2) to control the network device to operate as a slave on the LAN when the processor determines that a list manager is operating on the LAN, including controlling the network device to transmit an addressed packet to the list manager that includes the device address of the network device, and (3) to control the network device to operate as the list manager for the LAN when the processor determines that no list manager is operating on the LAN, including controlling the network device to maintain a list of device addresses and to respond to a requestor's request for device addresses by transmitting an addressed packet containing device addresses to the requestor.

31. Computer-executable process steps stored on a computer-readable medium, the computer-executable process steps to control a network device on a local area network (LAN) in which plural different network devices communicate over the LAN by transmitting broadcast rackets that are not addressed to any one particular network device and by transmitting addressed packets that are addressed to a network device corresponding to a device address contained therein, the network device being controlled to operate as a list manager which maintains a list of device addresses for the LAN, and being controlled to operate as a slave which provides a device address of the network device to a list manager, the computer-executable process steps comprising:

a determining step to determine whether a list manager is operating on the LAN by transmitting a broadcast packet from the network device to request a list manager and by waiting for a response for a predetermined period of time after the broadcast packet has been transmitted, said determining step determining that no list manager is operating on the LAN if a response to the broadcast packet is not received by the network device after the predetermined period of time, and determining that a list manager is operating on the LAN if a response to the broadcast packet is received by the network device within the predetermined period of time;

a controlling step to control the network device to operate as a slave on the LAN when the determining step determines that a list manager is operating on the LAN, including controlling the network device to transmit an addressed packet to the list manager that includes the device address of the network device; and

a controlling step to control the network device to operate as the list manager for the LAN when the determining step determines that no list manager is operating on the LAN, including controlling the network device to maintain a list of device addresses and to respond to a requestor's request for device addresses by transmitting an addressed packet containing device addresses to the requestor.

WEST

Generate Collection

L10: Entry 7 of 8 File: USPT Jul 3, 2001

DOCUMENT-IDENTIFIER: US 6256322 B1
TITLE: Bundling multiple network management packets

Detailed Description Text (34):
More specifically, in step S701, the network interface receives a network packet. Step S702 determines whether the packet is a network management packet or some other packet such as a device-specific packet. If the network packet is not a management packet, then flow branches to step S704 which processes the packet such as processing in accordance with device-specific application 35.

CLAIMS:

12. The method according to claim 11, further comprising determining whether the received packet is a device-specific packet or a network management packet.

20. The method according to claim 17, further comprising determining whether the same network management packet is a device-specific packet or a network management packet.

WEST

Generate Collection

L10: Entry 8 of 8 File: USPT Apr 21, 1998

DOCUMENT-IDENTIFIER: US 5742607 A
TITLE: Method and apparatus for controlling two way communication via disparate physical media

Abstract Text (1):
A method and apparatus for controlling two way communication via disparate physical media. A computer comprises a central processor, a forward channel interface, a return channel interface, and a main memory, each being coupled to a bus. The forward channel interface is further coupled to interrupt the central processor and coupled to receive a packet from a forward channel. The main memory contains an interrupt service routine comprising a first set of code for passing the packet to a routine for decapsulating the packet and a second set of code for passing a second packet to the return channel interface. The method comprises a computer transferring a packet from a forward channel interface to a main memory. The central processor analyzes the packet to determine if the packet is a data packet or a network management packet. If the packet is a network management packet, the central processor creates a response packet and passes the response packet to a return channel interface.

Brief Summary Text (16):
A method of processing a packet is also described. The method comprises a computer transferring the packet from a forward channel interface to a main memory. The central processor analyzes the packet to determine if the packet is a data packet or a network management packet. If the packet is a network management packet, the central processor creates a response packet and passes the response packet to a return channel interface. The method may also include passing a data packet created by a user application to the return channel interface.

Detailed Description Text (24):
The routines of the shim layer provide several mechanisms to support the broadband network protocol as well as mechanisms for packet encoding and control. An analysis mechanism 260 analyzes a portion of a packet to determine if it is a network management packet or a data packet. Data packets pass from the network driver to the link layer routines. Data packets which pass from the link layer are passed to a serial interface mechanism 280.

Detailed Description Text (33):
If, in the analysis step 320, the processor determines that the incoming packet is a network management packet, the receive interrupt service routine continues with a create response step 340. A buffer is allocated for a response packet which is encapsulated with status information if requested by the network management packet. One convenient method for buffer allocation is to maintain a circular queue. In an embodiment using Novell's ODI, an event control block (ECB) tracks a packet, thus a circular queue of ECBs is maintained.

CLAIMS:

1. A computer system comprising:

a central processor coupled to a bus;

a forward channel interface coupled to the bus, coupled to interrupt the central processor, and coupled to receive a packet from a forward channel which uses a first transmission media;

a return channel interface coupled to the bus, the return channel interface being adapted to transmit packets using a second transmission media different than the first transmission media:

a main memory coupled to the bus, the main memory containing a user application and an interrupt service routine, the interrupt service routine comprising a first set of code for passing the packet to a routine for decapsulating the packet, the main memory also containing a second set of code for passing a second packet to the return channel interface, and a third set of code for determining if the packet is a network management packet and calling the second set code to pass a response packet to the return channel interface if the packet is a network management packet.

11. A computer system comprising:

bus means for transferring computer system signals:

return channel interface means coupled to the bus means for returning packets using a return channel transmission media;

main memory means coupled to the bus means for storing a user application, a first routine for determining if an incoming packet is a network management packet, a second routine for passing a response packet to the return channel means if the incoming packet is a network management packet,

an interrupt service routine comprising a first set of code for passing the packet to a routine for decapsulating the packet, and a second set of code for passing a second packet to the return channel interface;

central processing means coupled to the bus means, the central processing means for executing the first and the second sets of code;

forward channel interface means coupled to the bus means, the forward channel interface means for receiving the incoming packet from a forward channel transmission media which is different than the return channel transmission media.

14. A method of processing a packet received from a service provider by a computer comprising a central processor, a main memory containing a user application, a forward channel interface, and a return channel interface, the method comprising the steps of:

(a) the computer interrupting the central processor from executing the user application upon transferring the packet which is received from a first transmission media by the forward channel interface to the main memory;

(b) the central processor analyzing the packet to determine if the packet is a data packet or a network management packet;

(c) if the packet is a network management packet, then the central processor responding to the packet by executing routines:

(i) creating a response packet;

(ii) passing the response packet to the return channel interface for transmission using a second transmission media.

18. A method of processing a packet received from a service provider via a first transmission media by a computer comprising a central processor, a main memory containing a user application, a forward channel interface, and a return channel interface, the method comprising the steps of:

(a) the computer transferring the packet from the forward channel interface to the main memory;

(b) the central processor entering a receive interrupt service routine from the user application;

(c) the central processor analyzing a predefined portion of said packet to determine if the packet is a data packet or a network management packet;

(d) if the packet is a network management packet then

(i) creating a response packet;

(ii) passing the response packet to the return channel interface for transmission to the service provider via a second transmission media.

WEST

Generate Collection

L18: Entry 4 of 8 File: USPT Nov 30, 1999

DOCUMENT-IDENTIFIER: US 5996010 A
TITLE: Method of performing a network management transaction using a web-capable agent

Brief Summary Text (18):
A network management function may involve the retrieval of network management information, the allocation of a value of a network management object, or the setting of a trap on a network management object.

CLAIMS:

6. The method of claim 5 including receiving a data packet at the network management agent, the data packet including a plurality of network management requests.

12. The method of claim 1 wherein the specific network management function is the setting of a trap on a network management object.

20. The computer-readable medium of claim 14 having stored thereon instructions which cause the processor to perform the specific network management function of setting a trap on a network management object.

WEST

Generate Collection

L18: Entry 6 of 8 File: USPT Aug 11, 1998

DOCUMENT-IDENTIFIER: US 5793975 A
TITLE: Ethernet topology change notification and nearest neighbor determination

Abstract Text (1):
It is desirable to be able to automatically map the topology of a computer network. To automatically map the topology of a computer network, a new method is proposed. First, all the network management modules (NMMs) in the network start off broadcasting multicast packets informing other units of their presence. When a network management module detects that only a single unit is connected to a particular slot-port combination, then that network management module designates the single unit as being a downstream unit in a network topology table. After updating its network topology table, the network management module sends a quench packet to the single unit to silence the downstream unit. These steps are repeated for all occurrences of a single unit connected to a particular slot-port combination in that network. After this occurs, the very bottom layer of the network has been detected and it's topology has been mapped. Since this bottom layer has been silenced by the quench packets, the bottom layer units will no longer be sending out the multicast packets. Thus, the next lowest layer can be detected by performing the same set of steps again. Specifically, any slot-pair combination that only has a single unit coupled to it is then marked as being a downstream and a quench message is sent to silence that unit. These steps are performed recursively until the entire network topology is detected.

Brief Summary Text (10):
To determine the network topology, every network management module (NMM) in an Ethernet network starts off sending multicast packets informing other network management modules of its presence. The multicast packets are SONMP packets that are defined in U.S. Pat. No. 5,226,120. When a network management module detects that only a single unit is connected to a particular slot-port combination, then that network management module designates the single unit as being a downstream unit in a network topology table. After updating its network topology table, the network management module sends a quench packet to the single unit to silence it. These steps are repeated for all occurrences of a single unit connected to a particular slot-port combination in that network. After this occurs, the very bottom layer of the network has been detected and it's topology mapped.

Detailed Description Text (15):
Having identified the leaf network management modules, the network management modules above the leaf units send "quench" packets to the leaf network management modules at step 240 to stop the leaf network management modules from transmitting multicast packets. The quench packets are standard ICMP packets that are well known within the Internet Protocol (IP). The step of instructing the lower network management modules to stop sending out multicast packets is known as "quenching". The quenched network management modules identify themselves as being downstream from the network management module that send the quench packet.

Detailed Description Text (26):
In the same fashion network management module C discovers that only network management module F is communicating to it through Slot 2/Port 4. Thus, network management module C marks network management module F as a downstream link through Slot 2/Port 4. Network management module C also sends a packet to network management module F to silence it as illustrated in FIG. 3b. Upon receiving the quench packet, network management module F modifies its topology table to indicate network management module B as being upstream.

Detailed Description Text (65):
In this situation, the quench packet will be received by network management module C at the new location and network management module C will send back a response. In the response sent by network management module C the port tagging of network management

module A will indicate that network management module C has moved from Slot 3/Port 1 of network management module A to Slot/3 Port 5 of network management module A. Network management module A will update this change of position in its topology table. To notify other units on the network, network management module A generates a Simple Network Management Protocol (SNMP) trap that informs other units on the network of the change.

Detailed Description Text (68):
When network management module G in FIG. 6 powers up it begins broadcasting multicast packets to all the other units connected to the network. Every network management module that receives the multicast packets will notice that network management module G to their topology table. This situation will be handled by step 461 of the receiving thread. Step 461 of the receiving thread clears the topology table, sets the DECENT-INTERVAL variable is to "false", and restarts the sender thread such that the network topology will be rebuilt.

Detailed Description Text (72):
In the case where network management module G has been powered-off, then network management module G will no longer respond to the quench packets periodically sent by network management module C. Since network management module C is no longer receiving response from its quench packets, network management module C will remove network management module G from its topology map. Network management module C also generates a Simple Network Management Protocol (SNMP) trap that will inform others of the topology change. The other network management modules will use the information in the trap to remove network management module G from their topology tables.

Detailed Description Text (74):
If the link between network management module C and network management module G remains down for a significant period of time then the set of events outlined in the previous paragraph will occur. Specifically, network management module C will remove network management module G from the its topology table and generate an SNMP trap that will inform others that network management module G is gone. However, if the link is brought back up then all the other units will begin to receive network management module G's multicast packets. The situation will therefore become that of FIG. 6 where network management module G has just been added to the network topology. Thus network management module G will be added to the topology as a new unit.